Taller Integrador: Patrones de diseño

# Sistema de Requerimientos Académicos.

# Objetivos

* Evaluar que patrones de diseño se deben utilizar en el desarrollo de un sistema.
* Aplicar varios patrones de diseño dentro de un mismo sistema.

# Requerimientos del sistema:

La empresa de facturación electrónica Stupendo lo ha contratado para desarrollar el sistema facturación de uno de sus clientes.

Este sistema debe ser capaz de emitir tres tipos de comprobantes electrónicos:

* Factura
* Guia de remision
* Nota de crédito

Entre los requerimientos solicitados por cliente se encuentran:

* Todos estos comprobantes electrónicos además del membrete obligatorio con los detalles del emisor deben poder personalizarse con el logo de su negocio y un lema al final.
* Debido a que actualmente el SRI maneja dos esquemas de facturación electrónica (offline y online) el cliente solicitó que al momento de pedir la autorización del comprobante al SRI, pueda escoger cualquiera de los dos esquemas para autorizar.

En el esquema Onine, el número de autorización es un nuevo código asignado por el SRI, mientras que en el esquema Offline el número de autorización es igual a la clave de acceso.

Las facturas tienen la siguiente estructura:

* Detalles del emisor
* Nombre del cliente
* Número de autorización
* Clave de acceso
* Código
* Fecha
* Lista de productos
* Total

Las guías de remisión tienen:

* Detalles del emisor
* Número de autorización
* Clave de acceso
* Fecha
* Destino
* Placa del vehículo de transporte
* Lista de productos

Las notas de crédito tienen :

* Detalles del emisor
* Número de autorización
* Clave de acceso
* Nombre del cliente
* Código de comprobante a modificar
* Fecha
* Detalle de modificación
* Valor a pagar

# Desarrollar

1. Indique que patrones podrían servir dentro del desarrollo de este sistema. (explique)
   1. Creacionales.

Factory Method: Debido a que las tres comparten características, son tipos de comprobantes electrónicos

* 1. Estructural.

Decorator.- permite añadir responsabilidades a un objeto dinámicamente, por ello las 3 tipos de comprobantes funcionan de manera distinta pero llevan algunos atributos iguales como el logo. Si posteriormente se le añadiese un nuevo componente, sería sencillo acoplarlo. además este patrón nos proporciona mayor flexibilidad que la herencia estática.

* 1. De Comportamiento

Strategy.- Esto permite que el algoritmo a ejecutarse varíe en función del cliente que lo use. En este caso el cliente va a elegir el tipo de comprobante. Este patrón define una familia de algoritmos, encapsula cada uno permitiendo el intercambio. De esta manera podremos añadir nuevos comportamientos o modificar

Diseñe un diagrama de clases del sistema

1. Implemente el sistema
   1. Cree un paquete por cada tipo de patrón

## En el programa principal muestre las funcionalidades del sistema.